

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Monitoring SSL spojení
Projekt do ISA

18. listopadu 2020

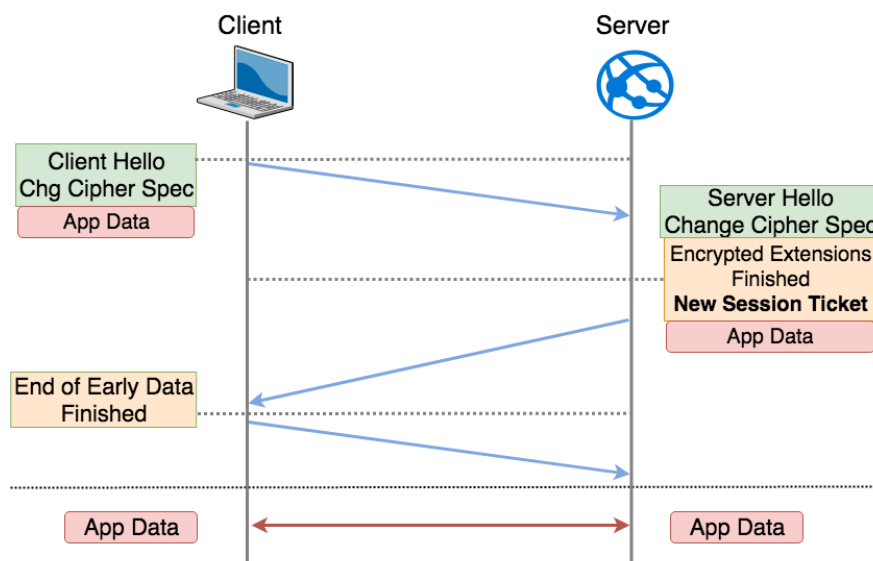
Michal Sova

Obsah

1	Úvod do problematiky	2
2	Implementace	2
2.1	SSL	2
2.2	Využití knihoven	3
3	Základní informace o programu	3
3.1	Návod na použití	3
3.2	Upřesnění chování	3
4	Použité zdroje	4
4.1	Obrázky	4

1 Úvod do problematiky

SSL/TLS vrstva se nachází z pohledu síťové architektury mezi aplikační a transportní vrstvou. SSL vrstva zajišťuje šifrování aplikační vrstvy. TLS spojení začíná Handshake. Handshake se skládá z Client a Server Hello. Klient posílá Client Hello obsahuje seznam podporovaných šifer, seznam kompresních metod a seznam rozšíření. V seznamu rozšíření lze většinou nalézt mimo jiné Server Name Indication (SNI). Server by měl poslat Server Hello, které obsahuje jen vybranou metodu šifrování a vybranou kompresní metodu. Součástí Handshaku by mělo být například certifikát, generování klíčů, Sever Hello Done.



Obrázek 1: SSL/TLS spojení[1]

2 Implementace

Program nejdříve zpracuje argumenty programu pomocí `getopts`[1]. Otevře rozhraní pro zachytávání paketů nebo otevře soubor se síťovým provozem, nastaví filtr na zachytávání tcp paketů a pomocí funkce `pcap_loop` a callback funkce[2] zachytává pakety. V callback funkci zavolá funkci `getAddress`. Tato funkce zjistí z IP nebo IPv6 hlavičky zdrojovou a cílovou adresu a velikost IP hlavičky. Z TCP se získá zdrojový a cílový port. Následně se spojení vyhledá ve vektoru `conn_vec`, v případě nenalezení se do vektoru vloží nové spojení. Následně se zjistí, zda-li jde o SSL spojení. V případě ukončení spojení se SSL spojení vypíše a spojení se smaže z vektoru spojení. Za ukončení spojení se považuje zachycení TCP s příznakem FIN z obou stran (od serveru i klienta), nebo zachycení TCP s příznakem RST z jedné strany.

2.1 SSL

Cyklus v callback funkci přečítá bajt po bajtu obsah a hledá příznaky SSL hlavičky, která obsahuje typ hlavičky, verzi protokolu a délku zprávy. Toto řešení je z důvodu možné různé velikosti obsahu tcp a velikosti SSL obsahu. V případě handshake program sleduje, zda se jedná o Client Hello nebo Server

Hello. V případě Client Hello se snaží najít Server Name Indication (SNI). V případě nenalezení se místo SNI nic nevypíše.

2.2 Využití knihoven

Bylo využito knihovny pcap.h a jejich funkcí pro zachytávání paketů. Knihovny netinet/ip.h pro použití struktury struct ip, ze které analyzátor načítá adresy a kontroluje další hlavičku. Podobně díky knihovně netinet/ip6.h a struktuře struct ip6_hdr. Z knihovny netinet/tcp.h bylo využito struktury struct tcphdr.

3 Základní informace o programu

Informace o SSL spojení se vypisují ve tvaru (na jednom řádku):

```
<timestamp>, <client ip>, <client port>, <server ip>, <SNI>, <bytes>, <packets>,  
<duration sec>
```

Všechny časové údaje jsou zaokrouhlené na 6 desetinných míst.

3.1 Návod na použití

- ./sslsniff -h pro zobrazení nápovědy.
- ./sslsniff -r file.pcapng kde file.pcapng obsahuje zachycený síťový provoz.
- ./sslsniff -i interface kde interface je rozhraní pro zachytávání. Tento příkaz je nutný zadat s dostatečnými právy (sudo).
- Nemá-li být uveden interface (avšak parametr -i je přítomen), vypíše se seznam aktivních rozhraní.

3.2 Upřesnění chování

- Program ignoruje neznámé argumenty.
- V případě zadání obou argumentů ('-r' i '-i'), program upřednostní čtení ze souboru (argument -r).
- Nemá-li ani jeden z parametrů '-r' a '-i' uveden, vypíše se nápověda.

4 Použité zdroje

Reference

- [1] Ashwin, V., 2015. How To Parse Program Options In C++ Using Getopt_Long. [online] Code Yarns. Dostupné z: https://codeyarns.com/2015/01/30/how-to-parse-program-options-in-c-using-getopt_long/
- [2] Carstens, T., 2020. Programming With Pcaptcpdump/LIBPCAP Public Repository. [online] Tcpdump.org. Dostupné z: <https://www.tcpdump.org/pcap.html>

4.1 Obrázky

Reference

- [1] AGA, 2018. 0RTT handshake state diagram. [online] Dostupné z: <https://labs.portcullis.co.uk/blog/playback-a-tls-1-3-story/#imageclose-6777>